PL-TR--94-1003

PL-TR-
94-1003

(1)

A FAST RECURSIVE ALGORITHM TO COMPUTE THE PROBABILITY OF M-OUT-OF-N EVENTS

AD-A283 679

George E. Radke, Jr.
Javon Evanoff

May 1994

Final Report

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

PHILLIPS LABORATORY
Directorate of Advanced Weapons and Survivability
AIR FORCE MATERIEL COMMAND
KIRTLAND AIR FORCE BASE, NM 87117-5776

PL-TR--94-1003

This final report was prepared by Phillips Laboratory, Kirtland Air Force Base, New Mexico, under Job Order 3647AB98. The Laboratory Project Officer-in-Charge was Captain Edward M. Jakes (WSAL).

This technical report has been reviewed and is approved for publication.

Edward M. Jakes

EDWARD M. JAKES, Capt, USAF
Project Officer

FOR THE COMMANDER

Corey D. King

COREY D. KING, Capt, USAF
Chief, Survivability/Vulnerability Branch

Brendan B. Godfrey

BRENDAN B. GODFREY, ST
Acting Director, Advanced Weapons and Survivability Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1994	Final; 1 Aug 93 - 1 Jan 94

4. TITLE AND SUBTITLE
A FAST RECURSIVE ALGORITHM TO COMPUTE THE PROBABILITY OF M-OUT-OF-N EVENTS

5. FUNDING NUMBERS
PE: 63605F
PR: 3647
TA: AB
WU: 98

6. AUTHOR(S)
G. E. Radke, Jr. and J. Evanoff*

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Phillips Laboratory
3550 Aberdeen Avenue SE
Kirtland AFB, NM 87117-5776

8. PERFORMING ORGANIZATION REPORT NUMBER
PL-TR--94-1003

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
*J. Evanoff is with Ball Aerospace, 2901 Juan Tabo NE, Albuquerque, NM 87112.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The successful performance of systems (power plants, aircraft, missiles, spacecraft, etc.) while degraded due to either man-made or natural stimuli is currently of interest to the survivability/vulnerability assessment communities. The fault tree model has long been used to qualify and quantify the likelihood of these degraded states. This report details the use of a particular recursive algorithm which is based on the 2-out-of-3 principle for M-out-of-N gates. Software currently available to calculate M-out-of-N gates is usually based on a cut set methodology for quantification. However, when N is too large, the number of cut sets becomes unmanageable. This algorithm provides a technique to calculate an exact probability when N is too large for cut set techniques.

14. SUBJECT TERMS
Statistical Analysis, Fault Tree Analysis, Fault Tolerance, Risk Assessment, Recursive Algorithm, Computer Aided Diagnosis, Degradation, Mathematics

15. NUMBER OF PAGES
14

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std Z39-18
298-102

CONTENTS

FIGURES

1.0 INTRODUCTION

The ability of systems (e.g., power plants, aircraft, missiles, spacecraft, etc.) to successfully perform their functions while degraded due to either man-made or natural stimuli is a subject of current interest by the probabilistic risk assessment and survivability/vulnerability assessment communities. A common tool used by these communities to both qualify and quantify the likelihood of these degraded states is the fault tree. Following the definition given by Barlow and Lambert (Ref. 1), a fault tree is a model that graphically and logically represents the various combinations of possible events occurring in a system that leads to the top event of interest. Figure 1 illustrates a simple fault tree for the top event of a disabled automobile air bag restraining system. The air bag system is disabled if either the inflation mechanism is disabled or both collision sensors are disabled. The combination of events in a fault tree is represented by special symbols which define logic gates where the most familiar gates are the AND and OR gates. The AND gate is passed if all of its inputs occur whereas the OR gate is passed if one or more of its inputs occur. The use of fault trees to graphically define the disablement logic of systems is widely accepted.

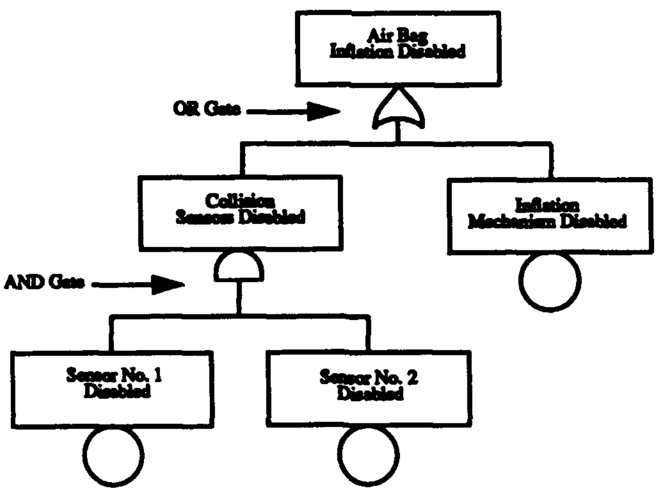

Figure 1. Example fault tree for automobile air bag restraining system.

Another less frequently used but useful gate is the M-out-of-N gate. This gate is passed if only M or more input events occur. Figure 2 illustrates the common fault tree symbol used to denote M-out-of-N gate. Common instances where an M-out-of-N gate is used is in voting systems. For example, to reduce the number of unnecessary and expensive shutdowns of a production process due to spurious signals, a system could be designed to shut down if two or more sensors out of a suite of three redundant sensors indicate a problem.

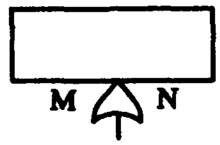

Figure 2. Common M-out-of-N gate symbol.

Interestingly, any M-out-of-N gate can be reduced to an equivalent set of AND and OR gates as illustrated in Figures 3 and 4 for a 2-out-of-3 gate. The convenience of using an M-out-of-N gate instead of its equivalent AND and OR gates becomes apparent when M and N become large.

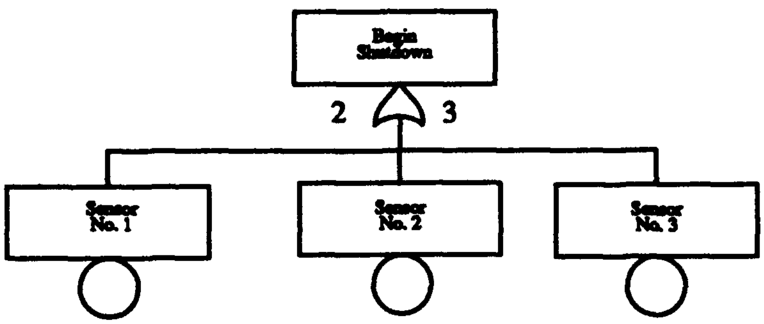

Figure 3. Example usage of a 2-out-of-3 gate.

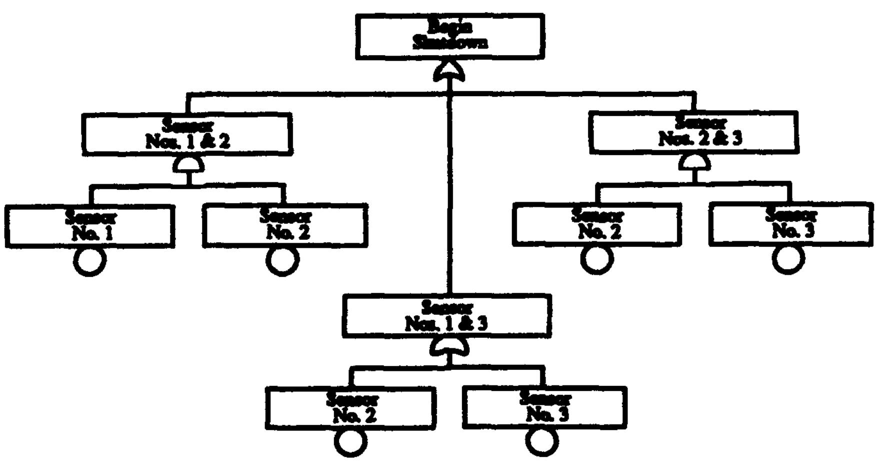

Figure 4. Equivalent representation of 2-out-of-3 fault tree shown in Figure 3.

Quantification of fault trees is a fairly straightforward process when probabilities are assigned to the input events and all the input events are statistically independent. The output of an AND gate is calculated by the multiplicative law of probability as defined in Equation 1 where Y_i denotes n input probabilities. Likewise, Equation 2 is a similar formula for calculating the output of an OR gate given n statistically independent input probabilities. By applying Equations 1 and 2 to the respective gates in a bottom-up fashion through the fault tree, the probability of the top event can be quantified.

$$\prod_{i=1}^{n} Y_i \tag{1}$$

$$1 - \prod_{i=1}^{n} [1 - Y_i] \tag{2}$$

The use of the M-out-of-N gates adds some additional complications in quantifying a fault tree. From inspection of Figure 4, it is clear that the assumption of statistical independence for all gate inputs is immediately negated. Consequently, quantifying an M-out-of-N gate requires knowledge of additional laws of probability followed by an exercise in the event-composition method of calculating the probability of an event.

In practice, most present day fault tree analysts do not quantify fault trees by hand, but rather use more sophisticated automated software tools (Refs. 2-4). Most of these tools are based on a cut set methodology for quantification. Basically, a cut set is a set of events whose occurrence causes the top event to occur. For example, the cut sets for the fault tree in Figure 4 are {Sensor No. 1, Sensor No. 2}, {Sensor No. 1, Sensor No. 3}, and {Sensor No. 2, Sensor No. 3}. For M-out-of-N gates where N becomes large, the number of cut sets becomes unmanageable even with a software tool. Equation 3 defines the number of cut sets which must be manipulated for an M-out-of-N gate which is simply the number of ways in which M objects can be selected out of N without regard to order.

$$\textit{Number of Cut Sets} = \frac{N!}{(N-M)!M!} \tag{3}$$

For example, for N = 25 and M = 10, there are 3,268,760 cut sets to be manipulated. To avoid this problem, many automated tools use an approximation or limit N to a relatively small number (e.g., 10). In most cases, the approximation is only valid when the input probabilities are very small (e.g., < 0.01).

For those situations where N is too large for cut set techniques and an exact probability is desired, the following algorithm is offered. This algorithm is easily coded into software for computational convenience.

Before proceeding with describing the algorithm, some notation and definitions are provided. Let

$e_1, e_2, ..., e_n$ represent N statistically independent events,

$p_1, p_2, ..., p_n$ are the probabilities of $e_1, e_2, ..., e_n$,

E(J,K) represents the event that exactly J of the K events $(e_1, e_2, ..., e_K)$ occurred,

P(J,K) is the probability of E(J,K),

$\cap$ represents Boolean AND operator,

$\cup$ represents Boolean OR operator, and

$\bar{e}$ denotes the negation of event e.

2.0 ALGORITHM

Suppose one has N statistically independent events $e_1, e_2, ..., e_n$ with known probabilities $p_1, p_2, ..., p_n$. Furthermore, let E(M,N) represent the event that exactly M of these N events occur, and let P(M,N) be the probability of E(M,N). Then find P(M,N) for M = 0, 1, ..., N. Since each p can be different, the binomial law does not apply.

Let E(J,K) be the occurrence of exactly J of the first K events $e_1, e_2, ..., e_k$, and let P(J, K) be the probability of E(J,K). Now it is shown that the probabilities P(J,K + 1), J = 0, 1, ..., K + 1 can be computed from the probabilities P(J,K), J = 0, 1, ..., K. Consider E(J, K + 1), the occurrence of exactly J of the first K + 1 events $e_1, e_2, ..., e_{k+1}$. Suppose J = 0. None of the first K + 1 events can occur only if none of the first K events occur, and event e_{k+1} also does not occur. This statement is written symbolically in Equation 4.

$$E(0,K+1) = E(0,K) \cap \bar{e}_{K+1} \tag{4}$$

Now compute the probability of the right side of Equation 4 to get an expression for P(0,K + 1), the probability of E(0,K + 1). The probability of E(0,K) is P(0,K) and the probability of $\bar{e}_{K+1}$ is $1 - p_{K+1}$. Since E(J,K) depends only on events e_1 through e_k which are statistically independent of e_{k+1} by definition, E(J,K) and e_{k+1} are independent for all values of J. Therefore, the simplified multiplicative law of probability can be used for the occurrence of two independent events as shown in Equation 5.

$$P(0,K+1) = P(0,K)\,[1 - p_{K+1}] \tag{5}$$

The special case of J = K + 1 is treated in a similar fashion. All of the first K + 1 events can occur only if all of the first K events occur, and event e_{k+1} also occurs. Equation 6 symbolizes this relation.

$$E(K+1,K+1) = E(K,K) \cap e_{K+1} \tag{6}$$

The corresponding probability of E(K + 1,K + 1) is shown in Equation 7.

$$P(K+1,K+1) = P(K,K)p_{K+1} \tag{7}$$

Now consider the cases where 0 < J < K + 1. For these cases, J of the first K + 1 events can occur in two ways. Either J of the first K events occur, and event e_{k+1} does not occur; or J - 1 of the first K events occur, and event e_{k+1} does occur. This is expressed symbolically in Equation 8.

$$E(J,K+1) = (E(J,K) \cap \bar{e}_{K+1}) \cup (E(J-1,K) \cap e_{K+1}) \tag{8}$$

Again invoking the independence of e_{k+1} from E(J,K) for all values of J, the probability of $(E(J,K) \cap \bar{e}_{K+1})$ is $P(J,K)(1 - p_{K+1})$, and the probability of $(E(J-1,K) \cap e_{K+1})$ is $P(J - 1, K)(p_{K+1})$. Since the two events $(E(J,K) \cap \bar{e}_{K+1})$ and $(E(J-1,K) \cap e_{K+1})$ are disjoint, the probability of either of them occurring is equal to the sum of their individual probabilities as shown in Equation 9.

$$P(J,K+1) = P(J,K)\,[1-p_{K+1}] + P(J-1,K)\,(p_{K+1}) \quad \text{for;} \tag{9}$$

$$(0 < J < K+1)$$

If P(J,K), J = 0, 1, ..., K, is known for some value of K = K', repeated use of Equations 5, 7 and 9 yields P(K,L), J = 0, 1, ..., K for all values of K greater than K' up to K = N. Since the values of P(J,K) are known for the trivial case of K = 1 ($P(0,1) = 1 - p_1$ and $P(1,1) = p_1$), the probabilities P(M,N), M = 0, 1, ..., N, can be computed in N-1 steps. Each step requires one application of both Equations 5 and 7, and K applications of Equation 9 where K is the step index. Therefore, the completion of P(M,N) for M = 0, 1, ..., N requires N - 1 applications of both Equations 5 and 7 and N(N - 1) / 2 applications of Equation 9. The individual probabilities P(M,N) can then be summed to get the probability of M or more events out of a possible N events.

3.0 EXAMPLE

Assume there are five statistically independent events, e_1, e_2, e_3, e_4, e_5 with known probabilities $p_1 = 0.20$, $p_2 = 0.50$, $p_3 = 0.30$, $p_4 = 0.90$, $p_5 = 0.70$. Now suppose one wants to calculate the probability of three or more of those events occurring. Using Equations 5, 7 and 9 in a recursive manner allows the answer to be computed as demonstrated in the following calculations.

$P(0,1) = 1 - p_1 = 1 - 0.20 = 0.80$
$p(1,1) = p_1 = 0.20$

$P(0,2) = P(0,1)[1 - p_2] = 0.80[1 - 0.50] = 0.40$
$P(1,2) = P(1,1)[1 - p_2] + P(0,1)p_2 = 0.20[1 - 0.50] + 0.80(0.50) = 0.50$
$P(2,2) = P(1,1)p_2 = 0.20(0.50) = 0.10$

$P(0,3) = P(0,2)[1 - p_3] = 0.40[1 - 0.30] = 0.28$
$P(1,3) = P(1,2)[1 - p_3] + P(0,2)p_3 = 50[1 - 0.30] + 0.40(0.30) = 0.47$
$P(2,3) = P(2,2)[1 - p_3] + P(1,2)p_3 = 0.10[1 - 0.30] + 0.50(0.30) = 0.22$
$P(3,3) = P(2,2)p_3 = 0.10(0.30) = 0.030$

$P(0,4) = P(0,3)[1 - p_4] = 0.28[1 - 0.90] = 0.028$
$P(1,4) = P(1,3)[1 - p_4] + P(0,3)p_4 = 0.47[1 - 0.90] + 0.28(0.90) = 0.299$
$P(2,4) = P(2,3)[1 - p_4] + P(1,3)p_4 = 0.22[1 - 0.90] + 0.47(0.90) = 0.445$
$P(3,4) = P(3,3)[1 - p_4] + P(2,3)p_4 = 0.030[1 - 0.90] + 0.22(0.90) = 0.201$
$P(4,4) = P(3,3)p_4 = 0.030(0.90) = 0.027$

$P(0,5) = P(0,4)[1 - p_5] = 0.028[1 - 0.70] = 0.0084$
$P(1,5) = P(1,4)[1 - p_5] + P(0,4)p_5 = 0.299[1 - 0.70] + 0.028(0.70) = 0.1093$
$P(2,5) = P(2,4)[1 - p_5] + P(1,4)p_5 = 0.445[1 - 0.70] + 0.299(0.70) = 0.3428$
$P(3,5) = P(3,4)[1 - p_5] + P(2,4)p_5 = 0.201[1 - 0.70] + 0.445(0.70) = 0.3718^*$
$P(4,5) = P(4,4)[1 - p_5] + P(3,4)p5 = 0.027[1 - 0.70] + 0.201(0.70) = 0.1488^*$
$P(5,5) = P(4,4)p5 = 0.027(0.70) = 0.019$

Summing all the probabilities which represent the occurrence of three or more events (those calculations with an asterisk) gives the probability of three or more events, 0.5396. The reader is encouraged to check the result.

4.0 SUMMARY AND CONCLUSIONS

This algorithm has been implemented into a software. Performance of the software is on the order of N^2 in time complexity and 2N in space complexity.

The order of N^2 in time complexity can be verified by observing that for any combination of M and N, the algorithm must compute the intermediate probabilities P(J,K) for all values of K, i.e., 1,2, ..., N. Furthermore, for each K, P(J,K) must be calculated on average of

$$\frac{1}{N}\sum_{i=1}^{N}(i+1) \qquad (10)$$

times to account for all J. Noting that

$$\frac{1}{N}\sum_{i=1}^{N}(i+1) \qquad (11)$$

is upper bounded by N, and neglecting the final linear summation of the appropriate intermediate probabilities, the computational complexity is $N \times N = N^2$.

The space complexity of 2N can be achieved by *recognizing* that the final intermediate probabilities P(J,N), J = 0, 1, ..., N depends only on the proceeding intermediate probabilities P(J,N - 1), J = 0, 1 ..., N - 1. Consequently, only two sets of intermediate probabilities must be kept on hand at any one time. Since the maximum number of intermediate probabilities is N + 1 (0, 1, ..., N), the total space requirement is N + (N + 1) which is approximately 2N.

A simple recursive algorithm is presented to compute the exact probability of occurrence of M or more events out of a possible N events. The algorithm begins by recursively commuting the probability of occurrence of exactly M events out of a possible N events. These intermediate results may be quickly summed to obtain the industry standard definition of M or more events out of a possible N events. Performance of the algorithm is on the order of N^2 in time and 2N in space. This algorithm does not use cut set methodology and consequently is not limited by the combinatorial explosion problem associated with cut set manipulation of the M-out-of-N gate.

This algorithm is extremely useful when the exact probability of M-out-of-N is desired, especially in cases where N exceeds limitations of cut set manipulation techniques and when the M-out-of-N event is statistically independent of other events in the system under consideration.

REFERENCES

1. Barlow, R. E., and Lambert, H. E. "Introduction to Fault Tree Analysis," *Reliability and Fault Tree Analysis Theoretical and Applied Aspects of System Reliability and Safety Assessment*, pp. 7-35, 1975.

2. Russell, K. D. et al., "Integrated Reliability and Risk Analysis System (IRRAS) Version 4.0 Reference Manual," NUREG/CR-5813, Nuclear Regulatory Commission, January 1992.

3. Koren, Jim and Riley, Jeff, CAFTA, Science Application International Corporation, Los Altos, California, September 1993.

4. Wakefield, Donald J., RISKMAN Probabilistic Risk Assessment, PLG, Inc., Newport Beach, California, September 1993.